In the 4/9/01 release

```
Public Function findCrystal( _
      ByVal imgPointer As Long, _
      ByVal pixResolution As Double, _
      ByRef resultArray() As Variant, _
      Optional ByVal displayResults As Boolean = True, _
      Optional ByVal minSize As Integer = -1, _
      Optional ByVal maxSize As Integer = -1, _
      Optional ByVal ROIcoordFlag As Double = 0 _
      ) As Boolean

    '***********************************************
    '*  findCrystal
    '*  created 3/26/01
    '*  by Mandel Mickley
    '*
    '*  This function returns the x,y position(s) of crystal(s) found within an image.
    '*  The position is returned as a dimensional offset from the center of the image.
    '*  Units for each position are in mm.
    '***********************************************

    Const PCT_MEAN = 1.1
    Const DILATE = 5
    Const ERODE = 10
    Const DARK_INT = 0              'intensity setting given to all pixels that are under the thresh
old
    Const LIGHT_INT = 255           'intensity setting given to all pixels that are above the thresh
old
    Const MAX_BLOBS = 255           'maximum number of blobs to hold data for
    Const DT_X = 20                 '# of x pixels per section of image to threshold
    Const DT_Y = 20                 '# of y pixels per section of image to threshold
    Const DT_SENS = 38              'percent of pixels in the region separated by a minimum RANGE
    Const DT_RANGE = 17             '# of gray levels separating high and low intensities
    Const DT_LEVEL = 51             'percent bias high or low of threshold
    Const DT_SMOOTHS = 1            '# of passes for smoothing of the sub region thresholds

    Dim imG As Long
    Dim img1 As Long
    Dim img2 As Long
    Dim imgTemp1 As Long
    Dim imgTemp2 As Long
    Dim Error As CInspLib_ErrorCodes
    Dim threshold As Double
    Dim i1 As Boolean
    Dim numBlobs As Integer         'number of blobs found
    Dim blbParms As Long            'pointer to structure for blob parameters
    Dim blbResults As Long          'pointer to structure fof blob results
    Dim grEnv As Long               'pointer to structure for graphics environment
    Dim imx As Integer              'size of image in the x direction
    Dim imy As Integer              'size of image in the y direction
    Dim blbx1 As Double             'left pixel pos of blob
    Dim blbx2 As Double             'right pixel pos of blob
    Dim blby1 As Double             'top pixel pos of blob
    Dim blby2 As Double             'bottom pixel pos of blob
    Dim bbperim As Double           'bounding box perimeter
    Dim bbdrop As Double            'bounding box perimeter of the drop
    Dim idrop As Integer            'index of blob representing the drop
    Dim xdrop As Double             'x position of drop
    Dim ydrop As Double             'y position of drop
    Dim dx1 As Double
    Dim dy1 As Double
    Dim dx As Double
    Dim dy As Double
    Dim i As Integer
    Dim b As Integer
    Dim value(8) As Variant
    Dim edgeBlobs As Integer
    Dim maxblob As Integer

    'threshold image
    If imgPointer <> 0 Then
```

Enclosure 7

```
        'create duplicate image to work with
        img1 = im_duplicate(imgPointer)
        img2 = im_dup(imgPointer)
        i1 = True
    Else
        findCrystal = False
        Error = RV_BAD_IMG_PTR
        Err.Raise Error, , "Invalid image pointer"
        Exit Function
    End If

    imgTemp1 = im_duplicate(img1)

    For i = 1 To ERODE
        If i1 Then
            Error = mvt_erode(img1, img2)
            i1 = False
            imG = img2
        Else
            Error = mvt_erode(img2, img1)
            i1 = True
            imG = img1
        End If
        If Error <> IM_OK Then
            Err.Raise Error, , "Failed dilation of image"
            GoTo errorLbl
        End If
    Next i

    For i = 1 To DILATE
        If i1 Then
            Error = mvt_dilate(img1, img2)
            i1 = False
            imG = img2
        Else
            Error = mvt_dilate(img2, img1)
            i1 = True
            imG = img1
        End If
        If Error <> IM_OK Then
            Err.Raise Error, , "Failed erosion of image"
            GoTo errorLbl
        End If
    Next i

    Error = mvt_thresh_st(imG, DT_X, DT_Y, DT_SENS, DT_RANGE, DT_LEVEL, DT_SMOOTHS)
    If Error <> IM_OK Then
        Err.Raise Error, , "Failed to threshold image"
        GoTo errorLbl
    End If

    'allocate blob structures
    blbParms = mvt_blob_create_params(Error)
    If Error <> IM_OK Then
        Err.Raise Error, , "Failed allocation of blob parameters"
        GoTo errorLbl1
    End If

    blbResults = mvt_blob_create_results(MAX_BLOBS, False, Error)
    If Error <> IM_OK Then
        Err.Raise Error, , "Failed allocation of blob results"
        GoTo errorLbl2
    End If

    'allocate graphics structures
    grEnv = gr_create_env(imG)
    If grEnv = 0 Then
        Error = RV_BAD_IMG_PTR
        Err.Raise Error, , "Could not generate graphics environment"
        GoTo errorLbl3
    End If

    'set blob parameters
    mvt_blob_set_min_area blbParms, minSize
    mvt_blob_set_max_area blbParms, maxSize

    'find blobs
    Error = mvt_blob_find(imG, blbParms, blbResults, ROIcoordFlag)
    If Error <> IM_OK Then
        Err.Raise Error, , "Failed blob analysis"
        GoTo errorLbl3
    End If

    'get the total number of blobs found
    numBlobs = mvt_blob_get_num_found(blbResults)

    ReDim resultArray(numBlobs, 8)

    imx = im_get_dx(imG) - 5
    imy = im_get_dy(imG) - 5

    For b = 0 To numBlobs

        blbx1 = mvt_blob_get_xfirst(blbResults, b)
        blbx2 = mvt_blob_get_xmax(blbResults, b)
        blbx1 = blbx1 - (blbx2 - blbx1)
        blby1 = mvt_blob_get_yfirst(blbResults, b)
        blby2 = mvt_blob_get_ymax(blbResults, b)

        If blbx1 > 5 Then
            If blbx2 < imx Then
                If blby1 > 5 Then
                    If blby2 < imy Then
                        bbperim = 2 * (blbx2 - blbx1) + 2 * (blby2 - blby1)
                            value(0) = bbperim
                            dx1 = blbx1
                            value(7) = dx1
                            dy1 = blby1
                            value(8) = dy1
                            dx = blbx2 - blbx1
                            value(5) = dx
                            dy = blby2 - blby1
                            value(6) = dy
                            xdrop = (dx / 2) + dx1
                            value(3) = xdrop
                            value(1) = (xdrop - (imx / 2)) * pixResolution
                            ydrop = (dy / 2) + dy1
                            value(4) = ydrop
                            value(2) = (ydrop - (imy / 2)) * pixResolution
                            sortAdd value, resultArray, False
                    Else
                        edgeBlobs = edgeBlobs + 1
                    End If
                Else
                    edgeBlobs = edgeBlobs + 1
                End If
            Else
                edgeBlobs = edgeBlobs + 1
            End If
        Else
            edgeBlobs = edgeBlobs + 1
        End If

    Next b

    gr_color grEnv, 128
    gr_circle imG, grEnv, resultArray(0, 3), resultArray(0, 4), 15
    gr_vectext imG, grEnv, resultArray(0, 3) + 20, resultArray(0, 4), 15, 0, Str(Round(resultArr
ay(0, 3))) + ", " + Str(Round(resultArray(0, 4)))
    gr_rectangle imG, grEnv, resultArray(0, 7), resultArray(0, 8), resultArray(0, 5), resultArra
```

```
y(0, 6)

    If displayResults Then
        mvt_copy imG, imgPointer
    End If

    'clean up
    'deallocate blob structures
    mvt_blob_delete_params blbParms
    mvt_blob_delete_results blbResults

    'deallocate graphics structures
    gr_delete_env grEnv

    im_delete img1
    im_delete img2
    im_delete imgTemp1

    'return results
    findCrystal = True

    Exit Function


errorLbl3:
    'deallocate graphics structures
    gr_delete_env grEnv
errorLbl2:
    'deallocate blob structures
    mvt_blob_delete_results blbResults
errorLbl1:
    'deallocate blob structures
    mvt_blob_delete_params blbParms

errorLbl:
    'delete images used for processing
    im_delete img1
    im_delete img2

    findCrystal = False

End Function
```